Vol.2017-CG-167 No.10
2017/9/19

XPBD: Position-Based Simulation of Compliant Constrained Dynamics 【CG技術の実装と数理 2017】研究報告

中川 展男[1,a)]

概要：筆者は Motion in Games 2016 で Miles Macklin らが発表した論文 “XPBD: position-based simulation of compliant constrained dynamics” [1] を追実装し【CG 技術の実装と数理 2017】で論文概要の紹介を行う予定である．この論文は, Position-Based Dynamics(PBD) [2] と呼ばれる手法にあったイテレーション回数やタイムステップの大きさによって剛性値 (stiffness) が変化してしまう問題を解決するために eXtended Position-Based Dynamics(XPBD) という手法を提案している．XPBD を用いることで、任意の弾性・散逸エネルギーポテンシャルを正確かつ効率的に陰解法で解くことが可能となり，さらに XPBD では、拘束力による力の推定が可能になるため切断可能ジョイント (breakable joints) や，触覚フィードバックが求められる場面などのより広い応用が可能となった．XPBD を用いることで PBD のシンプルさと堅牢性を保ったまま，より計算コストの高い非線形ニュートン解法と比較しても視覚的に遜色ない結果を生み出せる．

XPBD: Position-Based Simulation of Compliant Constrained Dynamics

“Mathematics and Implementation of Computer Graphics Techniques 2017” Technical Report

Nobuo Nakagawa[1,a)]

1. はじめに

1.1 背景と目的

Position-Based Dynamics(PBD) [2] は，ビデオゲームや映画，医療シミュレーションや VR などのインタラクティブなアプリケーションにおいて弾性体やクロスシミュレーションで広く知られる手法となってきている．この手法は，手法の持つシンプルさと堅牢さで魅力的だが，アプリケーションのタイムステップの大きさと，イテレーション回数に依存して拘束の剛性値 (stiffness) が変わってしまう問題が広く知られていた．特に，堅い剛体と柔らかい弾性体など異なる特性を持つオブジェクトが同時に存在する場合に，ある剛体の剛性値 (stiffness) を変更することで，弾性体の振る舞いも意図せず変わってしまうような場面では再利用可能なアセット作成が困難となっていた．また，たとえ単一のクロスシミュレーションであったとしても，クロスモデルの伸びと曲げの拘束を個別に変更したい場合にも独立した制御が困難になってしまう．さらにイテレーション回数の変更に応じて，振る舞いが非線形に変わるためパラメータを直感的に変更することが困難であった．

XPBD では，弾性ポテンシャルエネルギーのコンセプトに基づいた新しい拘束式を用いることで，PBD が持っていたタイムステップと，イテレーション回数依存で剛性値 (stiffness) が変わってしまう問題を解決している．また力に依存した効果やフォースフィードバックが可能なデバイスでの応用も可能となる．

1.2 関連研究

拘束系の力学シミュレーションの課題に取り組んだ多く

[1] 株式会社ポリフォニー・デジタル
Polyphony Digital Inc.
http://www.polyphony.co.jp
[a)] nakagawa@outlook.com

の先行研究があるが，XPBD は，Position Based Dynamics(PBD) [2] に基づいており，これは既存のオブジェクトに働く力を求めるアプローチとは異なり，直接位置を更新する手法である．近年は，拘束を解くのにグローバルな手法が用いられるようになってきているが，XPBD では 1 次微分の拘束導関数のみを用いタイムステップ毎のローカルな線形化を繰り返すことで，グローバル解法で生じているトポロジーが変更できない問題や事前計算が必要になる問題を回避している．

2. 手法

手法は下記で構成される．

1. 予測位置 $\tilde{x}$ を求める
2. 修正位置 x_0 を $\tilde{x}$ で初期化する
3. 乗数 λ_0 を 0 で初期化する
4. while (i < イテレーション回数)
   - 4.1 $\Delta\lambda$ を求める (式 18)
   - 4.2 Δx を求める (式 17)
   - 4.3 $\lambda_{i+1} = \lambda_i + \Delta\lambda$ で更新する
   - 4.4 $x_{i+1} = x_i + \Delta x$ で更新する
5. 位置 $x^{n+1} = x_i$ が確定する
6. 速度 $v^{n+1} = 1/\Delta t(x^{n+1} - x^n)$ で間接的に求まる

このうち，PBD と異なるのは，4.1 と 4.3 で計算するラグランジュ乗数の更新計算くらいで，私の方で追実装を行ったところほぼ計算コストの増加は見られなかった．つまり多くの場合で PBD とほぼ同一のコストでよりコントロール性の良い結果が得られるようになると考えて良い．

3. 実装

実装は C++ と OpenGL を用い Windows と macOS の両方の環境で動作を確認できるようにした．ベクトル演算等には glm ライブラリを用いた．筆者による実装のソースコードを github で公開している [4]．また関連論文 [1], [2] の日本語参考訳も筆者の承諾を得て同様に github で公開した [4]．また 9 月の研究発表会で，この実装例の実機デモを行う予定である．

実装結果のスクリーンショットを図 1 に示す．PBD と XPBD の手法の切り替えや，XPBD の現実の素材 (Concrete, Wood, Leather, Tendon, Rubber, Muscle, Fat) の切り替えも可能で，対話的に XPBD の有用性を理解できる内容になったと考えている．

4. 実験

論文で実験されている状況と同様の条件で筆者が独自に実装したシステムで再現実験をおこなったところ提供されている動画と同様に，PBD の持つ課題が解決され手法の有用性が確認できた．

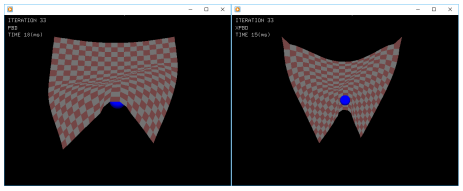

図 1 実装結果 (左:PBD，右:XPBD)

図 2 Slack の様子

5. おわりに

5.1 議論

実装にあたり，論文にも記載があるが XPBD は，繰り返し解法の収束を早める手法ではない事に注意が必要である．私が行った実験の範囲でも，極めて特殊なケースでは従来の PBD のほうが収束が早いケースがあった．ただし，XPBD は PBD に比べて，計算コストがほぼ同一のため，ほぼ全てのケースで XPBD のほうが有利になると考えられる．実際にすでに Obi Cloth 3.0 for Unity [3] などでの商用化の事例もあり，今後 PBD が用いられてきたケースの置き換えが進んでいくと考えられる．

昨年に引き続き 9 月の研究会までオンラインで議論を継続させるために参加者で構成されるプライベートな slack グループを作成した (図 2)．参加者は多くなかったが開発中の疑問点や各自の進捗を励みに自分の作業を進められたりなど良い効果があったと言える．

参考文献

[1] Miles Macklin, Matthias Müller, Nuttapong Chentanez.: *XPBD: position-based simulation of compliant constrained dynamics.*, MIG '16 Proceedings of the 9th International Conference on Motion in Games, Pages 49-54, (2016).
[2] Matthias Müller, Bruno Heidelberger, Marcus Hennix and John Ratcliff.: *Position based dynamics*, Journal of Visual Communication and Image Representation Volume 18 Issue 2, April, 2007, Pages 109-118, (2007)
[3] Obi Cloth 3.0 for Unity
入手先 ⟨http://blog.virtualmethodstudio.com/2017/01/obi-cloth-3-0-for-unity-whats-new/⟩ (2017.08.15).
[4] 筆者の github ページ, 入手先 ⟨https://github.com/nobuo-nakagawa⟩ (2017.08.15).